

The Aim of Backtesting - We are NOT trying to build the Ultimate Strategy

Aim of Backtesting

The aim of backtesting is to:

1. Check if our strategy performs well in periods when it is supposed to, and if it does not perform well in periods when it is not supposed to.
2. Understand how our strategy performs in different markets.
3. Get insights on how the strategy might be improved on.

1. Performance during selected periods

With a backtest, we can check to see if a strategy makes money when it is supposed to and loses money when it is supposed to.

For instance, let's say that our strategy is expected to perform better when the markets are volatile, or in other words, when they move much more than they normally do.

If our backtests then show that we make more money than expected during less volatile periods, this is a red flag (even though we made money).

We need to examine our strategy and figure out why.

2. Understand how the strategy performs in different markets

Different environments refer to different market conditions (high or low interest rates, trending, ranging, volatile, pandemic, and other crises etc) or on other assets.

Note that some strategies are meant for specific assets or specific asset classes. In this case, you generally want to check if strategy works for other assets within the same asset class.

After that is done, you can use your intuition to see how this strategy can be applied to other assets outside of the intended asset classes.

For instance, you can try our BAX strategy on other STIR futures. Next, you can think of trying it on other futures (commodities and cryptocurrencies).

By understanding the strength and weakness of our strategies, we can manage our strategies better when the market smacks us with different environments.

3. Improving the strategy

This involves making changes to a strategy after looking through the results of the backtest.

This is easily misused. You can use backtests to make small tweaks to your already viable (hopefully market prudent) strategies.

But in general, you should not be using backtests to come up with profitable strategies.

A common pitfall here is to continuously change the strategy so that it shows better results in a backtest. This usually leads to [overfitting](#) and [p-hacking](#). We will talk about those in the later lectures of this chapter.

This approach rarely leads to profitability when you trade it with real money.

The Ultimate Strategy

It is important to note that the aim of backtesting is not to build the ultimate strategy¹ that can withstand all market conditions.

We should not be trying to build something like that.

If you want to sell jewelry, would you try to sell meat, furniture and everything else along with it? No, you wouldn't. Selling more variety doesn't necessarily equate to higher overall sales. You don't try to be above average in everything, you want to focus on jewelry and do it very well.

Similarly, in trading, your strategy need not strive in all conditions. It just needs to do very well in one market condition, and as a trader, you deploy it when the market condition is right.

For instance, if your strategy performs in low volatility environments, turn it off or size down manually when an unexpected crisis like a pandemic occurs. You can then deploy another strategy that performs well in a high volatility environment.

Yes, your strategy should adapt to expected changes in volatility but when an extreme event happens, or if an environment that we know isn't good for our strategy happens, we need to step in.

Portfolio Robustness

Robustness means to strive in different conditions.

A single strategy might not do well in different conditions, but a portfolio of many strategies can.

Once you have many strategies, you can think of building 1) a well-rounded portfolio of strategies with 2) proper human intervention.

This way, in each period, you can use your discretion and deploy the strategies that you expect will work well and cut (or reduce the sizing) or those that aren't expected to do well.

Also, certain strategies play well with others and by deploying such strategies together, you can produce a higher return per risk.

You'll learn about portfolio design in later chapters.

That said, even with one strategy, you can survive well in many market conditions. Just deploy when the time is right and remove your strategy when it is not.

Ideally, you want a portfolio of strategies that are independent.

In addition, you want to design strategies that are sandboxed and well-hedged against unwanted risks.

¹ There isn't such a thing as an ultimate strategy but the bond futures calendar spread is close to being that.

Don't force a trade when there is none

Don't be concern about not having a portfolio of strategies.

Profitable strategies are hard to come by. I've only had 4 working strategies in my trading lifetime. There might be months of lull where nothing works. The key is to know when to let go.

As a retail trader, it is okay to not trade all the time. Forcing a trade equates to losing money. Losing money causes you to be at a lower starting point when you find your next working strategy.

Not trading when you don't have a working trading strategy is almost as important as trading when you have one.

But of course, during times when you are not running a profitable strategy, you should still be testing new ideas (on demo money or small amounts of real money).

In fact, you should always be testing new ideas, and improving old ones, even when you have a working strategy.

Having more strategies doesn't equal to more returns

If you have two equally good strategies, you need to split your capital among them. Hence, the overall returns will be the same.

That said, if the strategies are [independent](#), having more strategies does increase your overall reward per risk taken. Explanation by Ray Dalio (top hedge fund manager):
<https://www.youtube.com/watch?v=Nu4lHaSh7D4>

Conclusion: If possible, have multiple independent good strategies. If you can't find more than one good strategy, just stick to one and don't forcefully deploy other questionable strategies.